

A. CardSpace

1. Download the following to the c:\inetpub\wwwroot folder:

<http://buchananweb.co.uk/wwwroot.zip>

Enable Web Server

2. Initially we must enable the Web server on the machine. Once this is complete you should be able to access the local Web server with:

Control Panel -> Administrator Tools -> Computer Management ->

After this you should be able to test it locally from the browser with:

<http://localhost>

Are you able to access the Web server? Yes/No

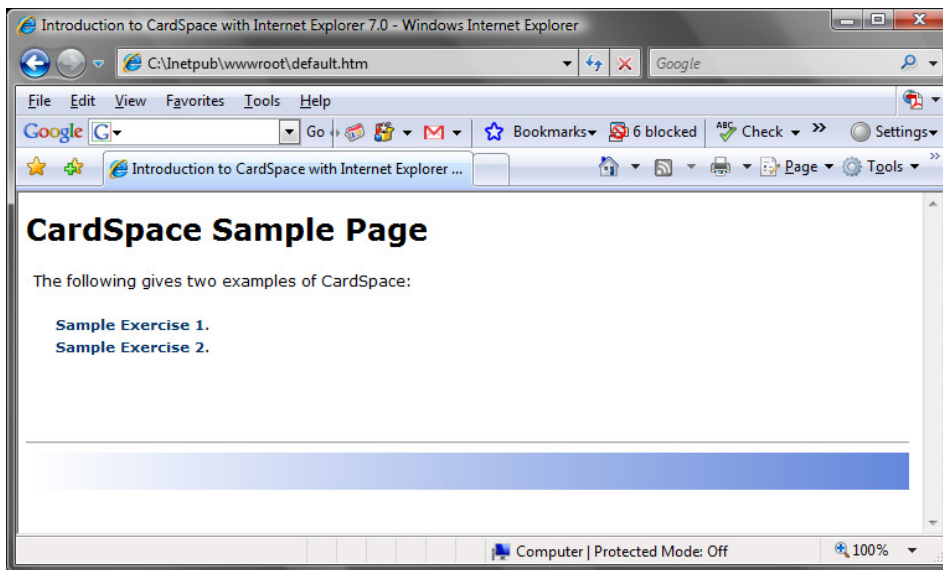


Figure 1:

Install Digital Certification for a Secure Connection

3. To enable CardSpace we need to install a digital certificate on the Web Server. To do this create a self-signed certificate by running:

Start->Microsoft Visual Studio 2008 -> Visual Studio Tools -> Visual Studio 2008 Command Prompt

Next create a certificate with (use you own name – replace Bill with your own name):

```
C:\Program Files\Microsoft Visual Studio 9.0\VC> makecert -n "CN=Bill" -ss MY -sr  
LocalMachine
```

Succeeded

Next determine the private key for the associated digital certificate:

```
C:\inetpub\wwwroot> findprivatekey.exe MY Localmachine -n "CN=Bill"
Private key directory:
C:\ProgramData\Microsoft\Crypto\RSA\MachineKeys
Private key file name:
1dd48849ebbe085453a464f9d7f49120_2fdb217-e3dd-446c-bbee-f2ade0616fd4
```

```
C:\Program Files\Microsoft Visual Studio 9.0\VC>certutil -store "my"

===== Certificate 6 =====
Serial Number: 0bfe8309707d5f9b4c4ff295233766d0
Issuer: CN=Root Agency
NotBefore: 12/11/2008 10:29
NotAfter: 31/12/2039 23:59
Subject: CN=Fred
Non-root Certificate
Cert Hash(sha1): 03 d9 df 4c b9 1b 97 a1 e3 5e 57 7d 15 24 78 72 25 16 53 f3
Key Container = JoeSoft
Unique container name: 7b90a71bfc56f2582e916a51aed6df9a_2fdb217-e3dd-446c-bbee-f2ade0616fd4
Provider = Microsoft Strong Cryptographic Provider
Private key is NOT exportable
Signature test passed
CertUtil: -store command completed successfully.
```

4. Next go to the home folder of the private key file (in this case c:\ProgramData\Microsoft\Crypto\RSA\MachineKeys).
5. Find the private key file, and right click on it, and add rights for ASPNET and the NETWORK SERVICE (see Figure 1).

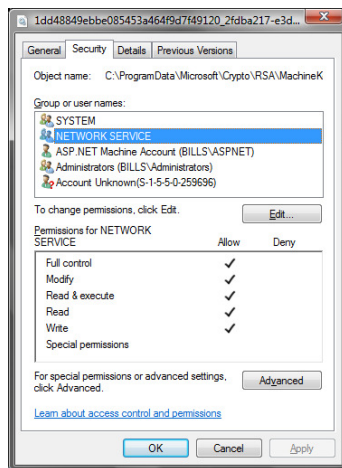


Figure 2: Settings the rights for the private key

- Next bind the certificate to HTTP by selecting the Default Web Site, and adding HTTPS (port 443), and binding the new certificate to it.

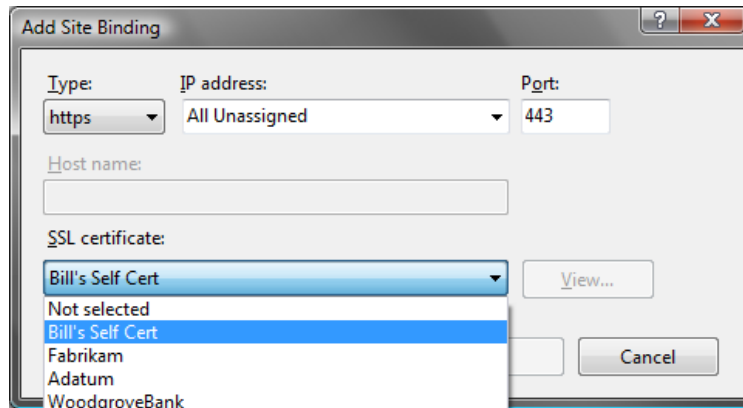


Figure 3: Setting the certificate (with Vista)

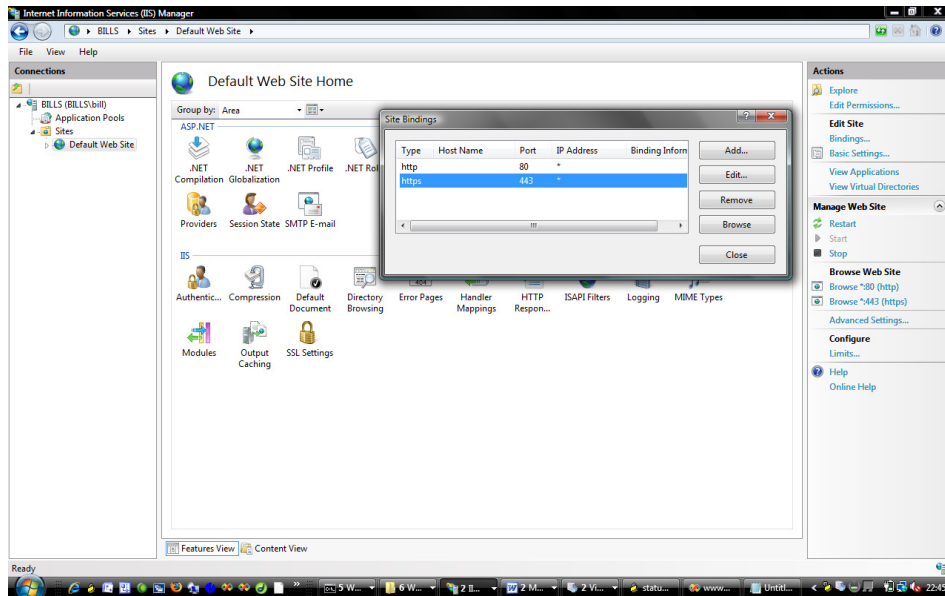


Figure 4: Adding the Binding

- Finally you should be able to make a secure connection to your local Web server, with <https://localhost>, such as:

Install Digital Certification for a Secure Connection

- To enable CardSpace we need to install a digital certificate on the Web Server. To do this create a self-signed certificate by running:

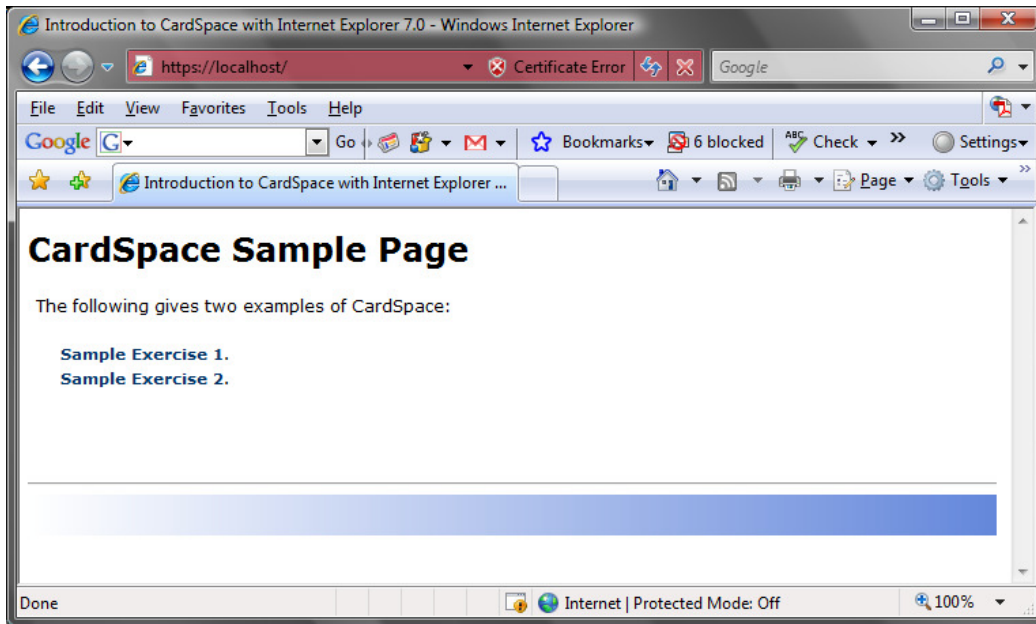


Figure 5: HTTPS access

As we have a self-signed certificate, the browser will not trust it (but it is okay to accept it).

Login with CardSpace (Part 1)

9. Next run Visual Studio 2008, and select Open Web site and navigate to c:\inetput\wwwroot.
10. Next select sample1.htm, and add the following code:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
<head>
  <title>Sample 1 </title>
</head>
<body>
  <form id="form1" method="post" action="cardspace1_login.aspx">
  <div>
    <button type="submit">Click here to sign in with your Information Card</button>
    <object type="application/x-informationcard" name="xmlToken">
      <param name="tokenType" value="urn:oasis:names:tc:SAML:1.0:assertion" />
      <param name="requiredClaims"
value="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/privatepersonalidentifier" />
    </object>
  </div>
  </form>
</body>
```

</html>

- Next select `cardspace1_login.aspx.cs`, and add the highlighted code:

```
protected void Page_Load(object sender, EventArgs e)
{
    Label1.Text = Request.Params["xmlToken"];
}
```

- Next load <https://localhost>, and select the first example (sample1.htm). Select your card (or create one), and login, such as:

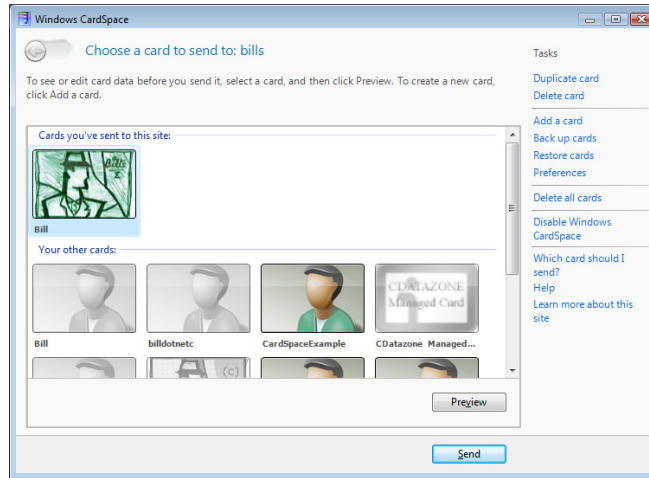


Figure 6: Cardspace selection

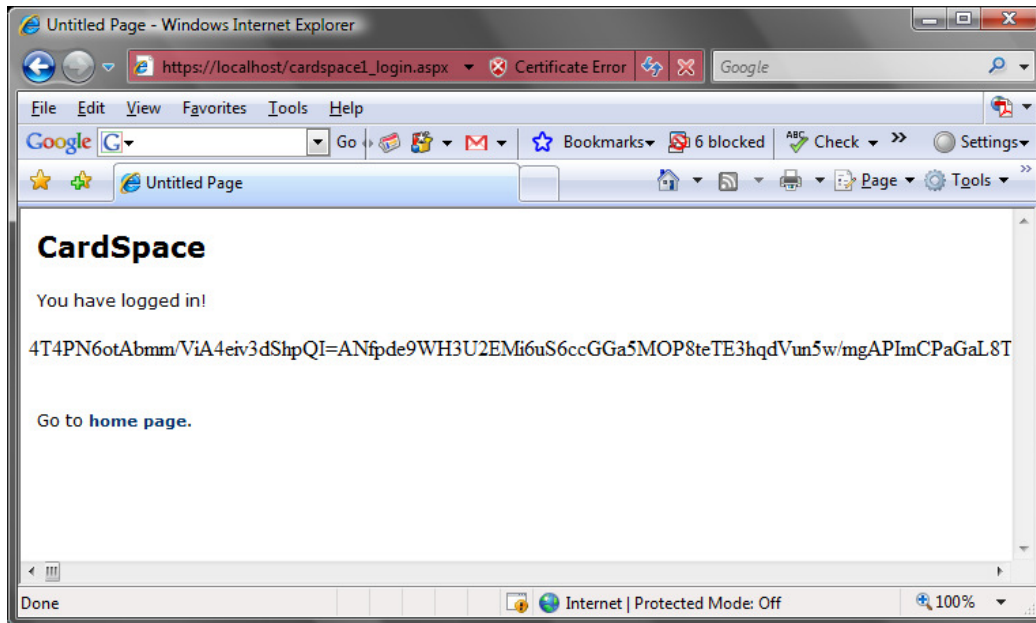


Figure 7: Cardspace login

Login with CardSpace (Part 2)

13. Next select sample2.htm, and add the following code:.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
<head>
  <title>Authenticate</title>
  <object type="application/x-informationcard" name="_xmlToken">
    <param name="tokenType" value="urn:oasis:names:tc:SAML:1.0:assertion" />
    <param name="requiredClaims"
value="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/privatepersonalidentifier" />
  </object>
  <script language="javascript">
    function GoGetIt()
    {
      var xmltkn=document.getElementById("_xmltoken");
      var thetextarea = document.getElementById("xmltoken");
      thetextarea.value = xmltkn.value ;
    }
  </script>
</head>
<body>
  <form id="form1" method="post" action="cardspace2_login.aspx">
  <div>
    <button name="go" id="go" onclick="javascript:GoGetIt();">Click here to get the token.</button>
    <button type="submit">Click here to send the card to the server</button>
    <textarea cols=100 rows=20 id="xmltoken" name="xmlToken" ></textarea>
  </div>
</form>
</body>
</html>
```

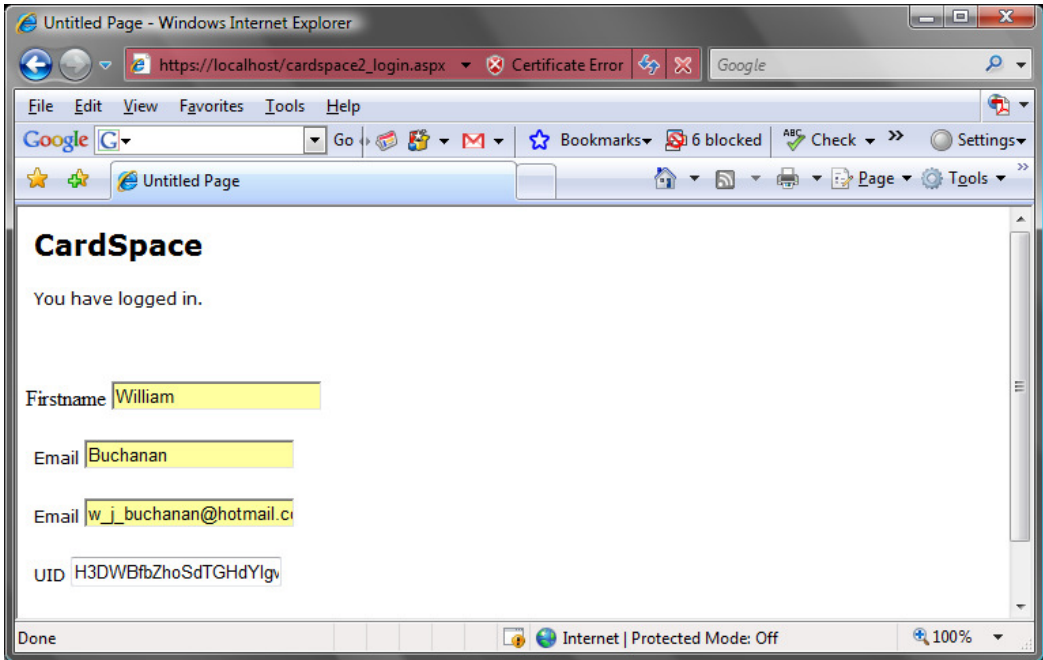
14. Next select cardspace2_login.aspx.cs, and add the highlighted code:

```
protected void Page_Load(object sender, EventArgs e)
{
  string xmlToken;
  xmlToken = Request.Params["xmlToken"];
  if (xmlToken == null || xmlToken.Equals(""))
  {
    // ShowError("Token presented was null");
  }
  else
  {

```

```
Token token = new Token(xmlToken);
firstname.Text = token.Claims[ClaimTypes.GivenName];
surname.Text = token.Claims[ClaimTypes.Surname];
email.Text = token.Claims[ClaimTypes.Email];
uid.Text = token.UniqueID;
}
}
```

Next show that the Web site now displays the details from the card, such as:



- 15. Finally, determine the IP address of a neighbouring machine, and access their machine with your card.

Note

If you want the final solution, download:

<http://buchananweb.co.uk/wwwrootfinal.zip>